

Analyse d'images

– Détection de caractéristiques –

Bibliographie

Ouvrages :

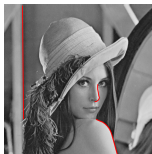
- *Digital Image Processing, 3rd Ed., chapter 11 "Representation and description"*, Rafael C. Gonzalez and Richard E. Woods, Prentice Hall, 2008.

Cours :

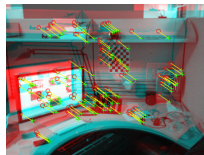
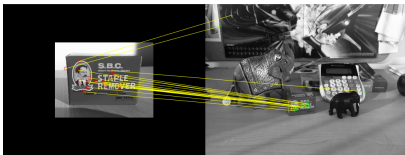
- Vincent Mazet, cours "Outils fondamentaux pour le traitement d'image",
<http://miv.u-strasbg.fr/mazet/ofti>
- Vincent Noblet, cours "Traitement d'images" TICS2A,
http://icube-miv.unistra.fr/fr/index.php/Traitement_d'images_TICS2A

Détection de caractéristiques

Objectif : rechercher les zones d'une image ayant des propriétés locales remarquables, par exemple un changement brusque et local d'intensité, un changement de texture, points particuliers, etc.



Intérêt : souvent utilisée en prétraitement en vue de résoudre un problème plus difficile (détection d'objets, interprétation, recalage, suivi d'objets dans une vidéo, etc).



Détection de caractéristiques

Caractéristiques :

- les contours,
- les lignes droites,
- les objets ou motifs d'intérêt,
- et bien d'autres (coins, formes, etc).

Propriétés attendues :

- robuste au bruit,
- robuste aux modifications d'acquisition,
- autres (selon application visée).

Plan du chapitre

1. Détection de contours

- 1.1 Différents types de contours
- 1.2 Modèles de contours luminance
- 1.3 Opérateurs différentiels
- 1.4 Robustesse au bruit
- 1.5 Gradient morphologique

2. Détection de lignes

3. Détection d'objets/motifs

Différents types de contours

On distingue différents types de contours :

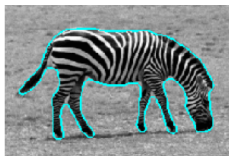
→ contours de luminance,



Différents types de contours

On distingue différents types de contours :

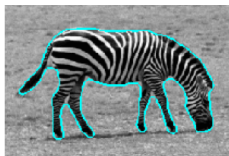
- contours de luminance,
- contours texture,



Différents types de contours

On distingue différents types de contours :

- contours de luminance,
- contours texture,
- contours couleur.



Outils pour la détection

Détection des contours de type luminance :

- filtrage (cf exemple TP 1)
- opérateurs différentiels et morphologiques

Outils pour la détection

Détection des contours de type luminance :

- filtrage (cf exemple TP 1)
- opérateurs différentiels et morphologiques

Détection des contours de type texture : → non traité dans ce cours

- analyses statistiques
- analyse spectrale (Fourier)

Outils pour la détection

Détection des contours de type luminance :

- filtrage (cf exemple TP 1)
- opérateurs différentiels et morphologiques

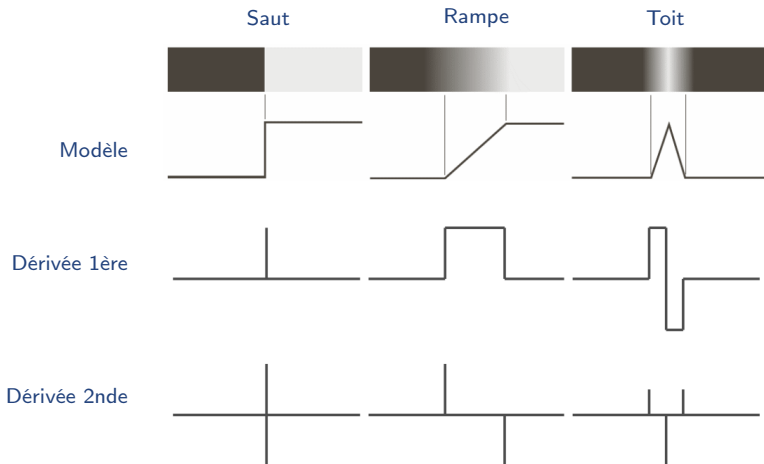
Détection des contours de type texture : → non traité dans ce cours

- analyses statistiques
- analyse spectrale (Fourier)

Détection des contours de type couleur : → non traité dans ce cours

- analyse couleur

Modèles de contours luminance (1D)



Modèles de contours luminance (1D)

Quelques remarques :

- La dérivée 1^{ère} permet de détecter la présence d'un contour en analysant son amplitude.
- La dérivée 2nde introduit deux ou trois valeurs pour chaque contour (non souhaité !)
- La dérivée 2nde permet de localiser le centre du contour.
- Ces remarques sont identiques quelle que soit l'orientation du contour (il suffit d'étudier le profil d'intensité perpendiculairement au contour).

Gradient et Laplacien

→ Dérivée 1^{re} (**gradient**)

$$\nabla I(x, y) = \begin{pmatrix} \frac{\partial I(x, y)}{\partial x} \\ \frac{\partial I(x, y)}{\partial y} \end{pmatrix}$$

→ Dérivée 2^e (**laplacien**)

$$\nabla^2 I(x, y) = \begin{pmatrix} \frac{\partial^2 I(x, y)}{\partial x^2} \\ \frac{\partial^2 I(x, y)}{\partial y^2} \end{pmatrix}$$

→ En pratique peu utilisé car très sensible au bruit.

Gradient et Laplacien

→ Dérivée 1^{re} (**gradient**)

$$\nabla I(x, y) = \begin{pmatrix} \frac{\partial I(x, y)}{\partial x} \\ \frac{\partial I(x, y)}{\partial y} \end{pmatrix} = \begin{pmatrix} G_x(x, y) \\ G_y(x, y) \end{pmatrix}$$

→ Dérivée 2^e (**laplacien**)

$$\nabla^2 I(x, y) = \begin{pmatrix} \frac{\partial^2 I(x, y)}{\partial x^2} \\ \frac{\partial^2 I(x, y)}{\partial y^2} \end{pmatrix}$$

→ En pratique peu utilisé car très sensible au bruit.

Gradient et Laplacien

→ Dérivée 1^{re} (**gradient**)

$$\nabla I(x, y) = \begin{pmatrix} \frac{\partial I(x, y)}{\partial x} \\ \frac{\partial I(x, y)}{\partial y} \end{pmatrix} = \begin{pmatrix} G_x(x, y) \\ G_y(x, y) \end{pmatrix} = \begin{pmatrix} I(x+1, y) - I(x, y) \\ I(x, y+1) - I(x, y) \end{pmatrix}$$

→ Dérivée 2^e (**laplacien**)

$$\nabla^2 I(x, y) = \begin{pmatrix} \frac{\partial^2 I(x, y)}{\partial x^2} \\ \frac{\partial^2 I(x, y)}{\partial y^2} \end{pmatrix}$$

→ En pratique peu utilisé car très sensible au bruit.

Gradient et Laplacien

→ Dérivée 1^{re} (**gradient**)

$$\nabla I(x, y) = \begin{pmatrix} \frac{\partial I(x, y)}{\partial x} \\ \frac{\partial I(x, y)}{\partial y} \end{pmatrix} = \begin{pmatrix} G_x(x, y) \\ G_y(x, y) \end{pmatrix} = \begin{pmatrix} I(x+1, y) - I(x, y) \\ I(x, y+1) - I(x, y) \end{pmatrix}$$

→ Dérivée 2^e (**laplacien**)

$$\nabla^2 I(x, y) = \begin{pmatrix} \frac{\partial^2 I(x, y)}{\partial x^2} \\ \frac{\partial^2 I(x, y)}{\partial y^2} \end{pmatrix} = \begin{pmatrix} I(x+2, y) - 2I(x+1, y) + I(x, y) \\ I(x, y+2) - 2I(x, y+1) + I(x, y) \end{pmatrix}$$

→ En pratique peu utilisé car très sensible au bruit.

Gradient d'une image – définition

Le gradient horizontal est défini par :

$$G_x(x, y) = I(x + 1, y) - I(x, y)$$

Gradient d'une image – définition

Le gradient horizontal est défini par :

$$G_x(x, y) = I(x + 1, y) - I(x, y) = \sum_i \sum_j H_x(i, j) I(x - i, y - j)$$

Gradient d'une image – définition

Le gradient horizontal est défini par :

$$G_x(x, y) = I(x + 1, y) - I(x, y) = \sum_i \sum_j H_x(i, j) I(x - i, y - j)$$

où :

$$\begin{cases} H_x(0, 0) = -1 \\ H_x(-1, 0) = +1 \\ H_x(i, j) = 0 \quad \text{ailleurs} \end{cases} \Rightarrow H_x = \begin{array}{|c|c|c|} \hline 0 & +1 & 0 \\ \hline 0 & -1 & 0 \\ \hline 0 & 0 & 0 \\ \hline \end{array}$$

Gradient d'une image – définition

Le gradient horizontal est défini par :

$$G_x(x, y) = I(x + 1, y) - I(x, y) = \sum_i \sum_j H_x(i, j) I(x - i, y - j)$$

où :

$$\begin{cases} H_x(0, 0) = -1 \\ H_x(-1, 0) = +1 \\ H_x(i, j) = 0 \text{ ailleurs} \end{cases} \Rightarrow H_x = \begin{array}{|c|c|c|} \hline 0 & +1 & 0 \\ \hline 0 & -1 & 0 \\ \hline 0 & 0 & 0 \\ \hline \end{array}$$

Le gradient vertical est défini par :

$$G_y(x, y) = I(x, y + 1) - I(x, y)$$

Gradient d'une image – définition

Le gradient horizontal est défini par :

$$G_x(x, y) = I(x + 1, y) - I(x, y) = \sum_i \sum_j H_x(i, j) I(x - i, y - j)$$

où :

$$\begin{cases} H_x(0, 0) = -1 \\ H_x(-1, 0) = +1 \\ H_x(i, j) = 0 \text{ ailleurs} \end{cases} \Rightarrow H_x = \begin{array}{|c|c|c|} \hline 0 & +1 & 0 \\ \hline 0 & -1 & 0 \\ \hline 0 & 0 & 0 \\ \hline \end{array}$$

Le gradient vertical est défini par :

$$G_y(x, y) = I(x, y + 1) - I(x, y) = \sum_i \sum_j H_y(i, j) I(x - i, y - j)$$

où :

$$\begin{cases} H_y(0, 0) = -1 \\ H_y(0, -1) = +1 \\ H_y(i, j) = 0 \text{ ailleurs} \end{cases} \Rightarrow H_y = \begin{array}{|c|c|c|} \hline 0 & 0 & 0 \\ \hline +1 & -1 & 0 \\ \hline 0 & 0 & 0 \\ \hline \end{array}$$

Opérateurs du gradient

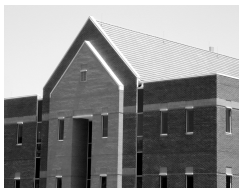
Détection de contours horizontaux et verticaux (définition) :

$$H_x = \begin{bmatrix} 0 & 1 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$H_y = \begin{bmatrix} 0 & 0 & 0 \\ 1 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

→ Il existe d'autres variantes de filtres pour approcher le calcul du gradient avec de meilleures propriétés (plus robuste au bruit, contours lisses, etc).

Image originale



Gx



Gy



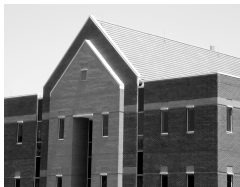
Opérateurs du gradient

Filtre de Robert (différence croisée = contour sur les diagonales) :

$$H_x = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

$$H_y = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$$

Image originale



Gx - Robert



Gy - Robert



Opérateurs du gradient

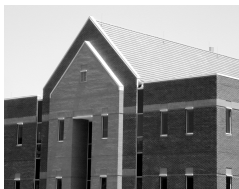
Filtres de Prewitt :

$$H_x = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

$$H_y = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$$

→ Symétrisation du calcul des dérivées horizontale et verticale.

Image originale



Gx - Prewitt



Gy - Prewitt



Opérateurs du gradient

Filtres de Sobel :

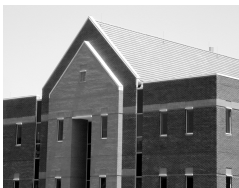
$$H_x = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

$$H_y = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

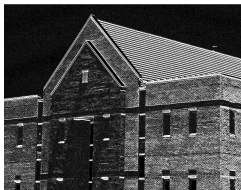
→ Plus robuste au bruit que Prewitt.

→ Contours plus lisses.

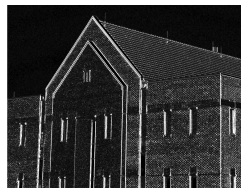
Image originale



Gx - Sobel



Gy - Sobel



Opérateurs du Gradient

- On peut aussi utiliser les masques de Prewitt et Sobel adaptés à la détection des contours dans la direction des diagonales

Filtres de Prewitt diagonaux :

$$H_x = \begin{bmatrix} 0 & 1 & 1 \\ -1 & 0 & 1 \\ -1 & -1 & 0 \end{bmatrix}$$

$$H_y = \begin{bmatrix} -1 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}$$

Filtres de Sobel diagonaux :

$$H_x = \begin{bmatrix} 0 & 1 & 2 \\ -1 & 0 & 1 \\ -2 & -1 & 0 \end{bmatrix}$$

$$H_y = \begin{bmatrix} -2 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 2 \end{bmatrix}$$

Des opérateurs de gradient à la détection de contours

Les différents opérateurs de gradients renvoient deux images (dérivées selon x et selon y ou selon les diagonales).

→ Comment passer à l'étape de détection de contours ?

Des opérateurs de gradient à la détection de contours

Les différents opérateurs de gradients renvoient deux images (dérivées selon x et selon y ou selon les diagonales).

→ Comment passer à l'étape de détection de contours ?

Magnitude (module) du gradient :

$$G(x, y) = \sqrt{G_x(x, y)^2 + G_y(x, y)^2} \approx |G_x(x, y)| + |G_y(x, y)|$$

Direction (angle) du gradient :

$$A(x, y) = \tan^{-1} \left(\frac{G_y(x, y)}{G_x(x, y)} \right)$$

Des opérateurs de gradient à la détection de contours

Les différents opérateurs de gradients renvoient deux images (dérivées selon x et selon y ou selon les diagonales).

→ Comment passer à l'étape de détection de contours ?

Magnitude (module) du gradient :

$$G(x, y) = \sqrt{G_x(x, y)^2 + G_y(x, y)^2} \approx |G_x(x, y)| + |G_y(x, y)|$$

Direction (angle) du gradient :

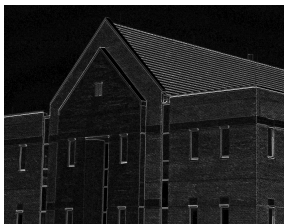
$$A(x, y) = \tan^{-1} \left(\frac{G_y(x, y)}{G_x(x, y)} \right)$$

→ Les contours sont les lieux où l'amplitude du gradient est maximale.

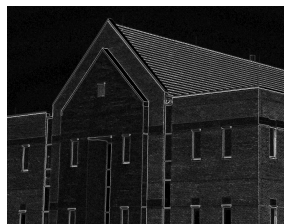
→ L'angle nous donne la direction (localement) perpendiculaire au contour.

Des opérateurs de gradient à la détection – exemples

G



G – Robert



G – Prewitt



G – Sobel



Des opérateurs de gradient à la détection

Détection :

→ Seuillage de l'image de magnitude du gradient.

Questions :

→ Choix du seuil ?

→ Que se passe-t-il en présence de bruit ?

Gradient seuillé à la valeur 50



Gradient seuillé à la valeur 100



Gradient seuillé à la valeur 200



Gradient seuillé à la valeur 250



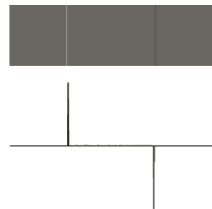
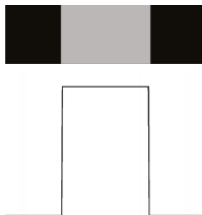
Impact du bruit sur la détection par gradient

Profil du contour

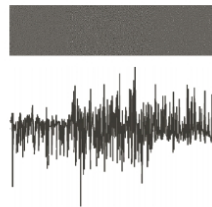
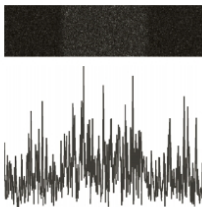
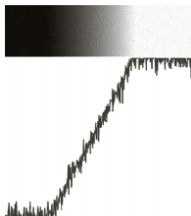
Dérivée 1ère

Dérivée 2nde

Sans bruit



Bruit



Impact du bruit sur la détection par gradient

Solution : appliquer les méthodes de débruitage sur l'image avant d'effectuer la détection de contour.

→ Lissage par convolution avec un noyau gaussien $H(x, y) = e^{-\frac{x^2+y^2}{2\sigma^2}}$.

Détecteurs robustes au bruit et aux erreurs de détection :

- Marr-Hildreth,
- Canny,
- etc.

Détecteur de Canny

Objectifs :

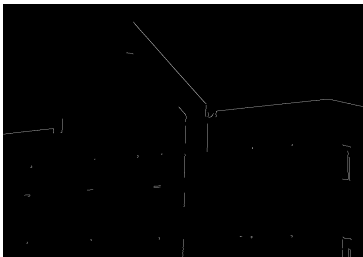
- tous les contours doivent être trouvés, avec un minimum de réponses parasites,
- les points du contours doivent être correctement localisés (la distance entre un point détecté et le vrai point du contour doit être la plus faible possible),
- un seul point doit être détecté pour chaque vrai point de contour.

Algorithme :

1. Lissage de l'image avec un filtre gaussien.
2. Calcul du gradient (magnitude et angle)
3. Suppression des non-maxima sur l'image de magnitude
4. Seuillage par hystérésis

Détecteur de Canny – Exemple avec différents seuils haut et bas

Détecteur de Canny, seuil bas : 0.5% et seuil haut : 0.6%



Détecteur de Canny, seuil bas : 0.1% et seuil haut : 0.6%



Détecteur de Canny, seuil bas : 0.1% et seuil haut : 0.3%



Détecteur de Canny, seuil bas : 0.1% et seuil haut : 0.2%



Gradient morphologique

Détection de contour sur les images binaires :

- Hypothèse : objets en blanc sur fond noir
- la détection de contour se fait avec un gradient morphologique défini par :

$$\frac{1}{2} ((I \oplus E) - (I \ominus E))$$

où E est un élément structurant.

Image seuillée



Gradient morphologique



Plan du chapitre

1. Détection de contours

2. Détection de lignes

2.1 Transformée de Fourier

2.2 Transformée de Hough

3. Détection d'objets/motifs

Détection de lignes par transformée de Fourier

Rappel : La transformée de Fourier met en évidence les structures présentes dans l'image.

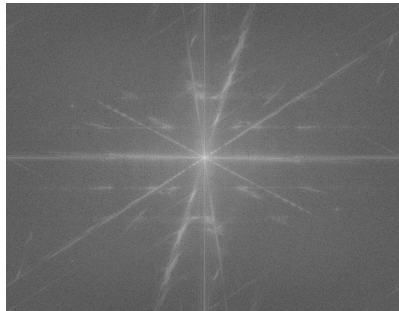
Question : Peut-on utiliser la transformée de Fourier pour détecter les lignes de l'image ?

Détection de lignes par transformée de Fourier

Image originale



Module de la transformée de Fourier



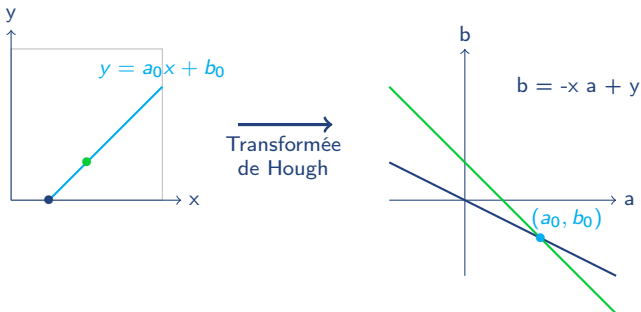
Le module donne l'orientation des lignes principales dans l'image : pas leur position !

Détection de lignes par transformée de Hough

La transformée de Hough est utilisée pour détecter des formes "simples" paramétrées par un petit nombre de paramètres.

→ exemple : une droite est paramétrée son coefficient directeur et son ordonnée à l'origine

Principe : changer d'espace de représentation pour passer dans l'espace des paramètres des formes que l'on cherche à détecter.

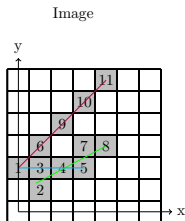


Détection de lignes par transformée de Hough – Algorithme

Principe :

1. Appliquer une détection de contour.
2. Définir un accumulateur = plan discrétisé des paramètres (a,b) .
3. Pour chaque point des contours :
 - Déterminer la droite correspondante dans l'espace des paramètres (a,b) .
 - Incrémenter l'accumulateur sur les points de cette droite.
4. Rechercher les maxima de l'accumulateur pour obtenir les paramètres des droites.

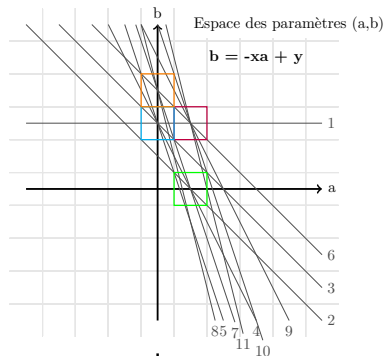
Détection de lignes par transformée de Hough – Exemple



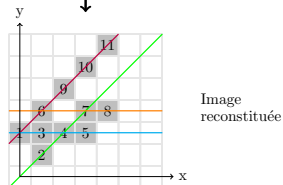
Coordonnées des points

- 1 : (0,2)
- 2 : (1,1)
- 3 : (1,2)
- 4 : (2,2)
- 5 : (3,2)
- 6 : (1,3)
- 7 : (3,3)
- 8 : (4,3)
- 9 : (2,4)
- 10 : (3,5)
- 11 : (4,6)

Transformée de Hough



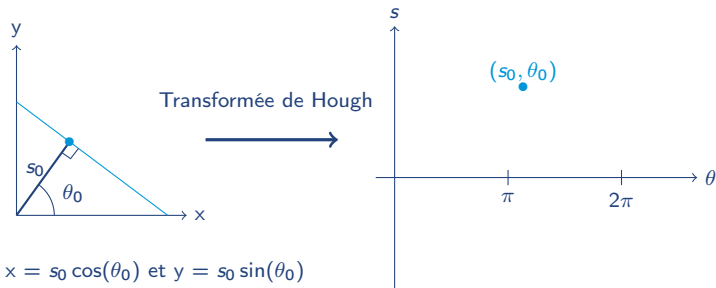
Transformée de Hough inverse



Détection de lignes par transformée de Hough – nouvelle paramétrisation

→ La paramétrisation (a,b) des droites ne permet pas la détection des droites verticales ($a = +\infty$)

Solution : Nouvelle paramétrisation



Détection de lignes par transformée de Hough – nouvelle paramétrisation

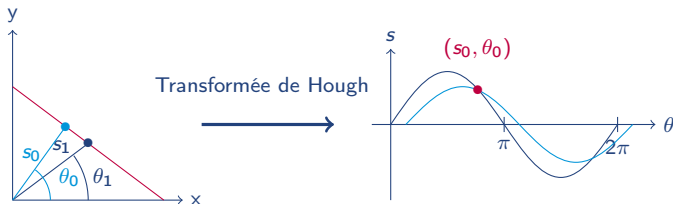
- A chaque point des contours, on peut associer un angle θ et une distance s avec la relation :

$$\begin{cases} x &= s \cos(\theta) \\ y &= s \sin(\theta) \end{cases}$$

- Un point de l'image correspond à une sinusoïde dans l'espace des paramètres $s = x \cos(\theta) + y \sin(\theta)$:

$$\begin{cases} x = s \cos(\theta) \\ y = s \sin(\theta) \end{cases} = \begin{cases} x \cos(\theta) = s \cos^2(\theta) \\ y \sin(\theta) = s \sin^2(\theta) \end{cases}$$

$$\Rightarrow x \cos(\theta) + y \sin(\theta) = s(\cos^2(\theta) + \sin^2(\theta))$$

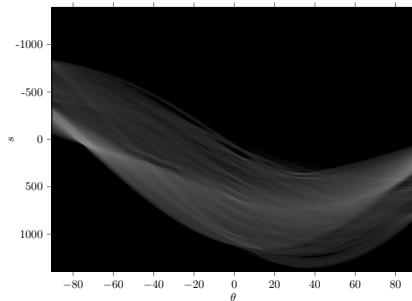


Détection de lignes par transformée de Hough – Exemple

Détecteur de Canny, seuil bas : 0.1% et seuil haut : 0.3%



Transformée de Hough, représentation (s, θ)



Détection de lignes par transformée de Hough

Avantages :

- Robuste au bruit.
- Robuste aux occlusions (peut détecter des objets partiellement recouverts.)
- Extensible à tout objet paramétré (cercle, ellipses, ...)

Exemple pour la détection des cercles :

$$(x - a)^2 + (y - b)^2 = r^2 \quad \rightarrow \quad 3 \text{ paramètres.}$$

Inconvénients :

- La dimension de l'accumulateur dépend du nombre de paramètres et de la résolution (échantillonnage de l'espace des paramètres) → le temps de calcul et la mémoire utilisée deviennent vite conséquents.
- Pas d'indication de début et de fin des lignes détectées (droites détectées, pas des segments).

Plan du chapitre

1. Détection de contours

2. Détection de lignes

3. Détection d'objets/motifs

3.1 Motivations

3.2 Détection par corrélation

Détection d'objets

Un grand nombre d'applications :

- Détection et reconnaissance des lettres et des chiffres (plaques d'immatriculation, tri du courrier, etc)
- Détection de visages (vidéosurveillance, contrôle dans les aéroports, etc).
- Détection de véhicules (images satellites, applications militaires).

Détection d'objets

Un grand nombre d'applications :

- Détection et reconnaissance des lettres et des chiffres (plaques d'immatriculation, tri du courrier, etc)
- Détection de visages (vidéosurveillance, contrôle dans les aéroports, etc).
- Détection de véhicules (images satellites, applications militaires).

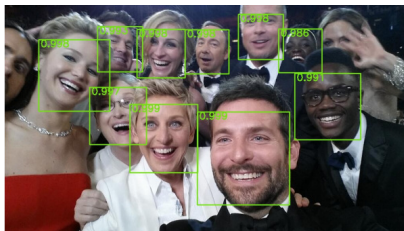
Pour détecter des objets, il faut :

- définir des caractéristiques représentatives des objets,
- que ces caractéristiques soient décrites par un petit nombre de paramètres,
- être en mesure de détecter ces caractéristiques (dans l'espace image ou dans l'espace des paramètres).

Détection d'objets – Visages

Détection de visages : algorithme de Viola & Jones

- Description des visages avec des caractéristiques nombreuses mais simples.
- Détection des caractéristiques (parcours de l'image et calcul optimisés).
- Phase d'apprentissage au préalable sur une base de photos de visages.
- Prise de décision par rapport à la phase d'apprentissage.
- Algorithme implémenté sur les appareils photos numériques.



Détection d'objets – Chiffres

Algorithme de détection et reconnaissances des chiffres :

- Caractérisation de la forme du chiffre (présence de cavités et direction de ces cavités).
- Analyse de la forme et décision avec un classifieur (souvent basé sur une base d'apprentissage).



Détection par corrélation

Corrélation normalisée :

→ La corrélation normalisée est définie par :

$$J(x, y) = \left(\frac{I}{\|I\|} \star \frac{H}{\|H\|} \right) (x, y) = \sum_i \sum_j \frac{H(i, j)}{\|H\|} \frac{I(x + i, y + j)}{\|I\|}$$

→ Mesure de ressemblance / mise en correspondance.

→ Le résultat $J(x, y)$ de la corrélation compris entre -1 et 1.

Détection par corrélation

Corrélation normalisée :

→ La corrélation normalisée est définie par :

$$J(x, y) = \left(\frac{I}{\|I\|} \star \frac{H}{\|H\|} \right) (x, y) = \sum_i \sum_j \frac{H(i, j)}{\|H\|} \frac{I(x + i, y + j)}{\|I\|}$$

→ Mesure de ressemblance / mise en correspondance.

→ Le résultat $J(x, y)$ de la corrélation compris entre -1 et 1.

Avantage :

→ Robustesse au bruit si le bruit est aléatoire et non corrélé au motif d'intérêt.

Détection par corrélation

Corrélation normalisée :

→ La corrélation normalisée est définie par :

$$J(x, y) = \left(\frac{I}{\|I\|} \star \frac{H}{\|H\|} \right) (x, y) = \sum_i \sum_j \frac{H(i, j)}{\|H\|} \frac{I(x + i, y + j)}{\|I\|}$$

→ Mesure de ressemblance / mise en correspondance.

→ Le résultat $J(x, y)$ de la corrélation compris entre -1 et 1.

Avantage :

→ Robustesse au bruit si le bruit est aléatoire et non corrélé au motif d'intérêt.

Inconvénients :

→ Il faut connaître l'image de l'objet à détecter.

→ Traitement long s'il y a beaucoup de motifs (orientations et tailles) à corrélérer et si les images sont grandes.

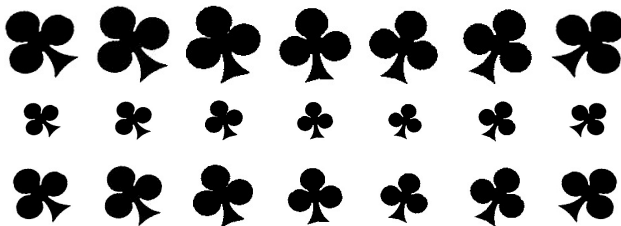
Détection par corrélation



→ Combien ai-je de cartes trèfle dans ma main ?

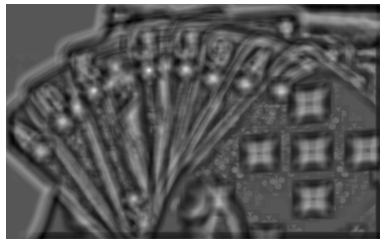
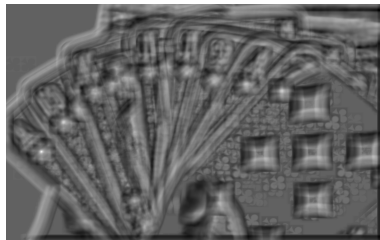
Détection par corrélation

- Corrélation normalisée avec une bibliothèque d'images contenant des trèfles orientés différemment avec différentes taille.



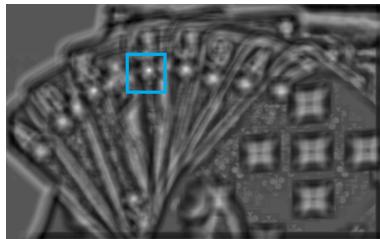
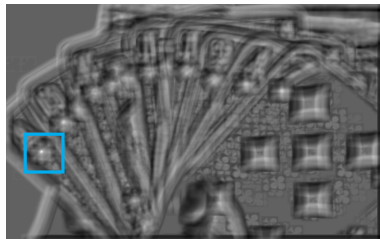
Détection par corrélation

→ Exemple de résultats



Détection par corrélation

→ Exemple de résultats



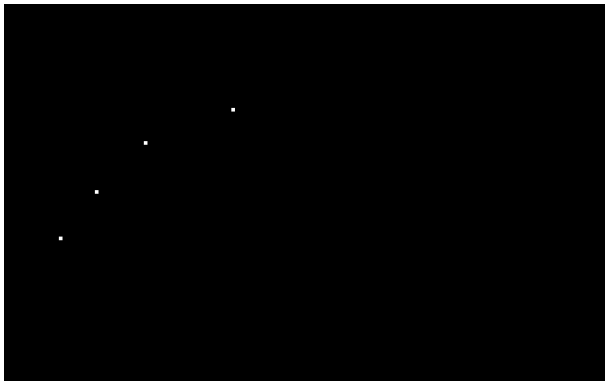
Détection par corrélation

→ Localisation des maxima de corrélation dans chaque résultat (corrélation ≥ 0.9).



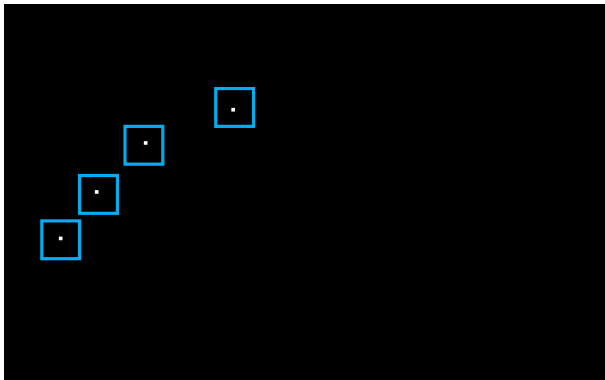
Détection par corrélation

→ Localisation des maxima de corrélation dans chaque résultat (corrélation ≥ 0.9).



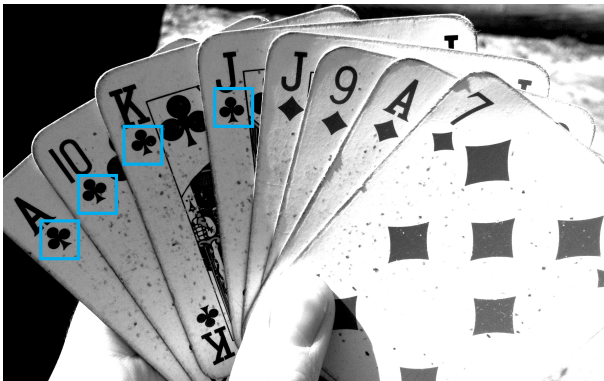
Détection par corrélation

→ Localisation des maxima de corrélation dans chaque résultat (corrélation ≥ 0.9).



Détection par corrélation

→ Localisation des maxima de corrélation dans chaque résultat (corrélation ≥ 0.9).



Conclusion

Bilan

- Détection de contours (filtrage, morphologie)
- Détection de lignes (transformée de Hough)
- Détection d'objets (corrélation)

Conclusion

Bilan

- Détection de contours (filtrage, morphologie)
- Détection de lignes (transformée de Hough)
- Détection d'objets (corrélation)

Bien d'autres caractéristiques existent :

- Caractéristiques basées sur les différences d'intensités
- Détection de coins (changement brusque d'intensité dans deux directions différentes)
- ...

Conclusion

Bilan

- Détection de contours (filtrage, morphologie)
- Détection de lignes (transformée de Hough)
- Détection d'objets (corrélation)

Bien d'autres caractéristiques existent :

- Caractéristiques basées sur les différences d'intensités
- Détection de coins (changement brusque d'intensité dans deux directions différentes)
- ...

Bien d'autres méthodes existent :

- Méthodes statistiques
- Apprentissage (machine learning)
- ...

A suivre ...

Recalage d'images